

Complexity of Steiner Tree in Split Graphs - Dichotomy Results

Madhu Illuri, P.Renjith, N.Sadagopan

Indian Institute of Information Technology, Design and Manufacturing, Kancheepuram, India.
 {coe11b012,coe14d002,sadagopan}@iiitdm.ac.in

Abstract. Given a connected graph G and a terminal set $R \subseteq V(G)$, *Steiner tree* asks for a tree that includes all of R with at most r edges for some integer $r \geq 0$. It is known from [ND12,Garey et. al [1]] that Steiner tree is NP-complete in general graphs. *Split graph* is a graph which can be partitioned into a clique and an independent set. K. White et. al [2] has established that Steiner tree in split graphs is NP-complete. In this paper, we present an interesting dichotomy: we show that Steiner tree on $K_{1,4}$ -free split graphs is polynomial-time solvable, whereas, Steiner tree on $K_{1,5}$ -free split graphs is NP-complete. We investigate $K_{1,4}$ -free and $K_{1,3}$ -free (also known as claw-free) split graphs from a structural perspective. Further, using our structural study, we present polynomial-time algorithms for Steiner tree in $K_{1,4}$ -free and $K_{1,3}$ -free split graphs. Although, polynomial-time solvability of $K_{1,3}$ -free split graphs is implied from $K_{1,4}$ -free split graphs, we wish to highlight our structural observations on $K_{1,3}$ -free split graphs which may be used in other combinatorial problems.

1 Introduction

Steiner tree is a classical combinatorial optimization problem which continues to attract researchers from both mathematics and computing. Interestingly, this problem finds applications in Network Design, Circuit Layout Design, etc., [3]. Given a connected graph G and a subset of vertices (terminal set) $R \subseteq V(G)$, Steiner tree asks for a tree spanning the terminal set. The objective is to minimize either the number of edges in the Steiner tree or the number of additional vertices ($Q \subseteq V(G) \setminus R$, also known as Steiner vertices). It is apparent from the definition that Steiner tree generalizes well-known Minimum Spanning Tree (MST) and Shortest Path problems in general graphs [4].

On the complexity front, Steiner tree in general graphs is NP-complete as there is a polynomial-time reduction from Exact 3 Cover [5]. Under the assumption, NP-complete problems are unlikely to have polynomial-time algorithms, it is natural to identify the gap between polynomial-time solvability and NP-completeness by restricting the input instances. Towards this end, many special graph classes such as chordal, bipartite, planar, split, etc., were discovered in the literature [6]. Classical problems such as Vertex cover, Clique, Odd-cycle transversal have polynomial-time algorithms when the input is restricted to chordal graphs which are otherwise NP-complete for arbitrary graphs [5]. However, other famous problems such as Hamiltonian Path (Cycle), Steiner tree, etc., remain NP-complete even on chordal graphs [2,7]. In fact, Steiner tree is NP-complete on Split graphs which are a strict subclass of chordal graphs [6]. Steiner tree is considered to be a difficult combinatorial problem compared to other problems as it is NP-complete on almost all special graph classes. For example, it is NP-complete on planar [8], chordal [2], bipartite [5], chordal bipartite [9] graphs. Due to its inherent difficulty, this problem has been an active research problem in the literature for the past three decades.

When a combinatorial problem is NP-complete on special graph classes such as chordal and split, it is natural to restrict the input further by means of forbidden subgraphs. For example, Hamiltonian cycle problem is NP-complete in chordal graphs, whereas it is polynomial-time solvable on interval graphs which are chordal and asteroidal-triple free [10,11,12,13]. In this paper, we revisit Steiner tree restricted to split graphs. It is known from [2], that Steiner tree on split graphs is NP-complete. We investigate the complexity of Steiner tree on subclasses of split graphs and present an interesting dichotomy. Towards this end, we study $K_{1,3}$ -free (claw free) and $K_{1,4}$ -free split graphs from both structural and algorithmic perspectives. In particular, we establish the following results;

- Steiner tree on $K_{1,5}$ -free split graphs is NP-complete.

- Steiner tree on $K_{1,4}$ -free split graphs is polynomial-time solvable.

Towards this end, we present a tight lower bound on the size of the Steiner set and our algorithm correctly produces such a Steiner set. The above results rightly identify the gap between NP-completeness and polynomial-time solvable input instances of Steiner tree problem restricted to split graphs. Since our contribution evolved from $K_{1,3}$ -free split graphs, we highlight structural results of both $K_{1,3}$ -free and $K_{1,4}$ -free split graphs. Although, the complexity of Steiner tree in $K_{1,3}$ -free split graphs is inferred from $K_{1,4}$ -free split graphs, out of combinatorial curiosity, we investigate both graphs from structural perspective and present polynomial-time algorithms for Steiner tree. To the best of our knowledge, this line of investigation has not been reported in the literature. The polynomial-time results known in the literature for Steiner tree are for trees and 2-trees [14].

As far as parameterized-complexity results are concerned, in [15] it is shown that Steiner tree in general is Fixed-parameter Tractable(FPT) if the parameter is the size of the terminal set and it is $W[2]$ -hard if the parameter is the size of the Steiner set [16]. From the domain of approximation algorithms, Steiner tree has a polynomial-time approximation algorithm with ratio $2 - \frac{1}{|R|}$ [17]. Variants of Steiner tree include Euclidean Steiner tree [18], Rectilinear Steiner tree [8], and Directed Steiner tree [19,20].

Roadmap: We present the structural characteristics of $K_{1,3}$ -free split graphs in Section 2. Using the structural observations made, we also present a polynomial-time algorithm to output a Steiner tree in $K_{1,3}$ -free split graphs. Structural characteristics of $K_{1,4}$ -free split graph and a polynomial-time algorithm to output a Steiner tree in $K_{1,4}$ -free split graphs is presented in Section 3. Hardness result is addressed in Section 4.

Graph-theoretic Preliminaries:

In this paper, we work with connected, simple, unweighted graphs. Notations are as per [6,21]. For a graph G the vertex set is $V(G)$ and the edge set is $E(G) = \{\{u, v\} \mid u, v \in V(G) \text{ and } u \text{ is adjacent to } v \text{ in } G \text{ and } u \neq v\}$. The neighborhood of vertex v is $N_G(v) = \{u \mid \{u, v\} \in E(G)\}$. The degree of a vertex v is $d_G(v) = |N_G(v)|$. $\delta(G) = \min \{d_G(v) \mid v \in V(G)\}$. For a graph G and $S \subseteq V(G)$, $G[S]$ represents the subgraph of G induced on the vertex set S . The subgraph relation is represented as $G[S] \subseteq G$. A *Split graph* $G = I + C$ is such that G can be partitioned into an Independent Set I and a Clique C , $V(G) = I \cup C$. A clique C is maximal if there does not exist a clique C' such that $C \subseteq V(C')$. For all split graphs mentioned in this paper we consider C to be a maximal clique unless otherwise stated. $K_{1,r}$ is a split graph on $r + 1$ vertices such that $|C| = 1$ and $|I| = r$, $E(K_{1,r}) = \{\{x, v\} \mid x \in C, v \in I\}$. $K_{1,3}$ is also termed as *claw*. *Centre vertex* of a $K_{1,r}$ is the vertex of degree r . A graph G is $K_{1,r}$ -free if G forbids $K_{1,r}$ as an induced subgraph. For a vertex $u \in C$, $N_G^I(u) = N_G(u) \cap I$ and $d_G^I(u) = |N_G^I(u)|$. For $S \subseteq C$, $N_G^I(S) = \bigcup_{v \in S} N_G^I(v)$, and $d_G^I(S) = |N_G^I(S)|$. For a split graph G , $\Delta_G^I = \max\{d_G^I(v) \mid v \in C \text{ and } V_3 = \{u \in C \mid d_G^I(u) = 3\}\}$. Two edges e_1 and e_2 are non adjacent if they do not share an end vertex in common. A set of edges $M \subseteq E(G)$ forms a matching of G if every pair of edges in M are non adjacent. Maximum matching is a matching of maximum cardinality in G . $\alpha(G)$ denotes the size of the maximum matching in G .

2 $K_{1,3}$ -free Split Graphs: Structural Results

In this section, we analyze the structure of $K_{1,3}$ -free split graphs and we present some interesting structural results. Further, we show that for a claw-free split graph G , if $\Delta_G^I = 2$, then $|I| \leq 3$. This acts as a good handle in yielding a linear-time algorithm for Steiner tree problem which we see in the later half of this section.

Theorem 1. *Let G be a connected split graph. G is claw free if and only if one of the following conditions hold.*

1. $\Delta_G^I \leq 1$
2. $\Delta_G^I = 2$ and for every $u, v \in C$ such that $d_G^I(u) = 2$, $N_G^I(u) \cap N_G^I(v) \neq \emptyset$

Proof. Necessity: Suppose $\Delta_G^I \geq 3$, and let $v \in C$ has at least 3 neighbours, say $x, y, z \in I$. Then the set $\{v, x, y, z\}$ forms a claw in G with v as its centre vertex. It follows that if G is claw-free, then $\Delta_G^I \leq 2$. Now suppose $\Delta_G^I = 2$. Let $u \in C$ such that there exist vertices $x, y \in I$, $\{x, y\} \subseteq N_G^I(u)$. We assume on the contrary that there exist $v \in C, v \neq u$ such that $N_G^I(u) \cap N_G^I(v) = \emptyset$. Since C is a clique, $\{u, v\} \in E(G)$. It follows that vertices $\{u, x, y, v\}$ forms a claw in G with u as its centre, a contradiction. This proves Condition

2, and completes the proof of the forward direction.

Sufficiency: On the contrary assume that G is not claw free. No claw in G can have its centre vertex in the set I , since for any v in I , the set $N_G(v) \subseteq C$ and hence induces a clique in G . So every claw in G has its centre vertex in the set C . Consider a claw with the vertex set $\{v, x, y, z\}$, with the centre v being in C . No two of the other three vertices of the claw can be in C , because then there would be an edge between them. So at most one of $\{x, y, z\}$ is in C , and the rest (of which there are at least two) are in I . It follows that if G contains a claw, then $\Delta_G^I \geq 2$. Equivalently, if $\Delta_G^I \leq 1$ then G is claw-free. Finally, consider the case where $\Delta_G^I = 2$. Suppose the vertex set $\{v, x, y, z\}$ induces a claw in G , with its centre vertex being v . Then v is in C , and at least two of $\{x, y, z\}$ are in I , as we argued above. Since $\Delta_G^I = 2$ we get that exactly two of $\{x, y, z\}$, say x and y , are in I . Then z is in C , and $\{x, z\}, \{y, z\} \notin E(G)$. It follows that $N_G^I(v) \cap N_G^I(z) = \emptyset$ which is a contradiction to Condition 2. Therefore, our assumption that there exist a claw in G is wrong, and this completes the sufficiency. Therefore, the theorem follows. \square

Lemma 1. *For a claw-free split graph G , if $\Delta_G^I = 2$, then $|I| \leq 3$.*

Proof. Since $\Delta_G^I = 2$, let there exist a vertex $v \in C$ such that $d_G^I(v) = 2$. On the contrary, assume that $|I| > 3$, that is, $\{a, b, c, d\} \subseteq I$ such that $N_G^I(v) = \{a, b\}$. Let $X = N_G(a)$ and $Y = N_G(b)$ as shown in Figure 1. If there exist a vertex $t \in C$ such that $t \notin X$, and $t \notin Y$, then vertices $\{a, b, v, t\}$ induces a claw. Therefore, $C = X \cup Y$. If $X \subseteq Y$, then $C \cup \{b\}$ induces a larger clique, which is a contradiction to the assumption on the maximality of clique C . Therefore, $X \not\subseteq Y$ and similarly, $Y \not\subseteq X$. It follows that, $X - Y \neq \emptyset$ and $Y - X \neq \emptyset$. For every vertex $v \in X \cap Y$, $\{v, c\} \notin E(G)$ and $\{v, d\} \notin E(G)$ otherwise, $N_G^I(v) \cup \{v\}$ induces $K_{1,3}$. Therefore, the vertices c, d can have adjacency in two disjoint sets $X - Y$ or $Y - X$.

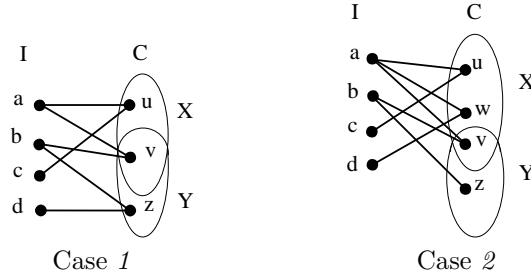


Fig. 1. An illustration for the proof of Lemma 1

Case 1: $N_G(c) \cap X \neq \emptyset$ and $N_G(d) \cap Y \neq \emptyset$. Edge $\{u, c\} \in E(G)$ where $u \in X - Y$ and $\{z, d\} \in E(G)$ where $z \in Y - X$. Observe that $\{a, z\}, \{c, z\} \notin E(G)$ otherwise $N_G^I(z) \cup \{z\}$ induces $K_{1,3}$. Similarly, $\{d, u\} \notin E(G)$ otherwise $N_G^I(u) \cup \{u\}$ induces $K_{1,3}$. From the discussion, it follows that the vertices $\{u, a, c, z\}$ induces a claw, which is a contradiction. Similar argument holds for $N_G(c) \cap Y \neq \emptyset$ and $N_G(d) \cap X \neq \emptyset$.

Case 2: $N_G(c) \subseteq X$ and $N_G(d) \subseteq X$. Let $\{c, u\}, \{d, w\} \in E(G)$ such that $u, w \in X - Y$. Note that there exist at least one vertex $z \in Y - X$. If $\{c, z\} \notin E(G)$, then the vertices $\{u, a, c, z\}$ induces a claw. If $\{c, z\} \in E(G)$, then the vertices $\{w, a, d, z\}$ induces a claw as, $\{d, z\} \notin E(G)$. The argument is symmetric for $N_G(c) \subseteq Y$ and $N_G(d) \subseteq Y$.

Cases 1 and 2 give a contradiction to the fact that G is claw free. Therefore, our assumption that $|I| > 3$ is wrong, and hence, the lemma follows. \square

2.1 Application: Steiner tree in $K_{1,3}$ -free Split Graphs

Using the structural results presented in Section 2, in this section, we present a polynomial-time algorithm to find minimum Steiner tree in $K_{1,3}$ -free split graphs. Optimum version of Steiner tree problem is defined as follows;

OPT Steiner tree(G, R)
 Instance: Graph $G(V, E)$, Terminal Set $R \subseteq V(G)$
 Question: Find a minimum cardinality set $S \subseteq V(G) \setminus R$ such that $G[S \cup R]$ is connected?

We here consider the Steiner tree problem on split graph $G^0 = I^0 + C^0$. Due to pruning, we iteratively construct split graphs G^1, G^2 from the input graph $G = G^0$. We simplify the input by pruning the vertices

which are not part of any optimum solution. The pruned graph G^1 is the graph induced on the vertex set $V(G^0) \setminus (S_1 \cup S_2 \cup S_3)$. Clearly, $G^1 \subseteq G^0$ and let $G^1 = I^1 + C^1$. We prune three sets of vertices S_1, S_2, S_3 one after the other and are defined as follows. $S_1 = \{a \in I^0 \mid a \notin R\}$. $S_2 = \{u \in C^0 \mid u \notin R \text{ and } N_G^{I^0}(u) \cap R = \emptyset\}$. Let $R' = \{v \in C^0 \mid v \in R\}$. $S_3 = \bigcup_{v \in R'} \{v\} \cup N_G^{I^0}(v)$. Consider the Steiner tree optimization problems P_1 ,

and P_2 defined as follows.

P_1 : OPT Steiner tree(G^0, R)

P_2 : OPT Steiner tree($G^1, R \setminus S_3$)

Lemma 2. *An optimum solution Q to P_2 is also an optimum solution to P_1 .*

Proof. Note that the first two sets S_1, S_2 pruned from G^0 are not part of any optimum solution. $S_3 \subseteq R$ induces a connected subgraph of G^0 which is also pruned to obtain G^1 . If $V(G^1) \cap R = \emptyset$, then Steiner set of P_2 is empty. i.e., R induces a connected subgraph of G^0 . On the other hand if $V(G^1) \cap R \neq \emptyset$, then there exist at least one vertex $v \in C^1$ in the Steiner set Q of P_2 . $Q \subseteq C^1$ connects all terminal vertices $R \setminus S_3$. If $S_3 \neq \emptyset$, then there exist at least one vertex $u \in S_3$ such that $u \in C^0$ and $u \in R$. $\{u, v\} \in E(G^0)$ and therefore, $Q \cup R$ induces a connected subgraph of G^0 and Q is a minimum Steiner set for P_1 . Hence, the lemma follows. \square

2.1.1 A polynomial-time algorithm to find a minimum Steiner tree Given a $K_{1,3}$ -free split graph G^0 with terminal vertex set $R \subseteq V(G^0)$, we present a polynomial-time algorithm to find a minimum Steiner tree. As part of preprocessing step, we prune the sets S_1 , and S_2 , which are not part of any optimum solution. Further, we delete terminals which are in C , and their neighbours in I , namely the set S_3 . Now we have an instance of Steiner Tree in claw-free split graphs where all the terminals are in the independent set. An optimum solution to the pruned graph is also an optimum solution to the original graph by the previous lemma. We now present a sketch of algorithm and the detailed one is presented in Algorithm 1. If $\Delta_G^I = 0$, then the instance is trivial. If $\Delta_G^I = 1$, then Steiner set should contain one neighbor vertex in C of each terminal in I . In the remaining case, $\Delta_G^I = 2$ and therefore, by Lemma 1 $|I| \leq 3$. The only non-trivial case is when $|I| = 3$. From the constraints of the instance, we know that it is necessary and sufficient to pick exactly two Steiner vertices from C in this case.

Algorithm 1 Steiner tree in Claw free Split graphs. Steiner_tree(G^0, R)

```

/* $G^0$  is a claw-free split graph and  $R \subseteq V(G^0)$  is the set of terminal vertices */
1: Find the pruned graph  $G^1 = \text{Pruning}(G^0, R)$ 
2: Initialize the output set of Steiner vertices  $S = \emptyset$  and unmark every vertices in  $I^1 \subseteq V(G^1)$ 
3: if  $\Delta_{G^1}^I = 1$  then
4:   for every unmarked vertex  $d \in I^1$  do
5:     include  $w \in C^1$  in  $S$  where  $\{d, w\} \in E(G^1)$ .
6:   mark vertex  $d$ .
7:   end for
8: else
9:   include vertex  $x \in C^1$  in  $S$  where  $|N_{G^1}^I(x)| = 2$ . i.e.,  $N_{G^1}^I(x) = \{a, b\}$ 
10:  if  $|I^1| = 3$ . i.e.,  $I^1 = \{a, b, c\}$  then
11:    include  $y \in C^1$  in  $S$  where  $\{c, y\} \in E(G^1)$ 
12:  end if
13: end if
14: Run standard Breadth First Search in the graph  $G[S \cup R]$  and output the BFS tree.

```

Algorithm 2 Pruning the input instance of Steiner tree. Pruning(G^0, R)

```

/*  $G^0$ : input claw-free split graph,  $R$ : set of terminal vertices */
1: Find the sets  $S_1, S_2, S_3$  in order and prune those vertices from  $G^0$ . i.e.,  $G^1 = G^0 \setminus S$  where  $S = S_1 \cup S_2 \cup S_3$ 
2: Return the pruned graph  $G^1$ .

```

2.1.2 Proof of Correctness of Algorithm 1

By Lemma 2, a minimum Steiner set of pruned graph G^1 is an optimum Steiner set for G^0 . Therefore, pruning in Step 1 is a solution preserving operation. We present a case analysis to show that our algorithm outputs a minimum Steiner tree of a claw-free split graph.

Case 1: $\Delta_{G^1}^I \leq 1$. Note that for every vertex $d \in I^1$, Step 5 includes exactly one vertex $w \in N_{G^1}(d)$ in S , which is a minimum Steiner set.

Case 2: $\Delta_{G^1}^I = 2$. Observe $|I| \leq 3$ by Lemma 1. $|S| = 1, 2$ for $|I| = 2, 3$, respectively, which is done by Steps 9, 11. Therefore, S is a minimum Steiner set for G^1 , and by Lemma 2, S is also a minimum Steiner set for G^0 . Step 14 outputs a Steiner tree by running standard Breadth First Search algorithm on $G[S \cup R]$.

2.1.3 Run Time Analysis

We represent the input claw-free split graph using an adjacency list, as we can easily find a neighbor of a given vertex. Vertices in adjacency list are arranged such that C^0 follow I^0 . Intuition behind this ordering is that, first neighbor of a vertex $v \in C^0$ encountered in the list is always a vertex $u \in I^0$, if it exists. If $\Delta_{G^1}^I = 1$, then $u \in N_{G^1}(v)$ can be determined in constant time. Therefore, Algorithm 1 takes linear time $O(n)$, $n = |V(G^0)|$ to output a minimum Steiner set.

3 $K_{1,4}$ -free Split Graphs: Structural Results

In this section, we first analyze the structure of $K_{1,4}$ -free split graphs. Subsequently we investigate Steiner tree problem restricted to $K_{1,4}$ -free split graphs. Towards this end, we give a nice bound on the cardinality of any minimum Steiner set. Further, we present a structural characterization of $K_{1,4}$ -free split graph meeting the bound. Interestingly, the characterization yields a polynomial-time algorithm to output a minimum Steiner tree, which we shall present in Section 3.1.

Before we present the structural results, we introduce some additional terminologies. A split graph G is a l -split graph if $\Delta_G^I = l$. Note that a $K_{1,4}$ -free split graph is a l -split graph for some $l, 0 \leq l \leq 3$, and the converse does not always hold. In a split graph G , closed neighborhood of a vertex $u \in C$ is $[N(u)] = \{u\} \cup N_G^I(u)$. For a l -split graph $G = I + C, 0 \leq l \leq 2$, we construct a *labeled* graph M such that $V(M) = I$ and $E(M) = \{\{a, b\} \mid a, b \in I \text{ and } N_G(a) \cap N_G(b) \neq \emptyset\}$ and label the edge $\{a, b\}$ as v_{ab} . Note that v in v_{ab} denotes a vertex $v \in N_G(a) \cap N_G(b)$. Also, we pick exactly one $v \in N_G(a) \cap N_G(b)$ to label the edge $\{a, b\}$. For any edge set $E^* \subseteq E(M)$, we define the *corresponding vertex* set V^* as follows. Corresponding to each edge $\{a, b\} \in E^*$, include exactly one vertex $v \in N_G(a) \cap N_G(b)$ in V^* . It follows that, $V^* \subseteq C$ and $|V^*| = |E^*|$. Clearly, $|V^*| \leq |E^*|$ as we are including not more than one vertex in V^* corresponding to each edge in E^* . Suppose $|V^*| < |E^*|$, then there exist at least two edges labelled v_{ab}, v_{cd} in E^* such that $v \in N_G(a) \cap N_G(b)$ and $v \in N_G(c) \cap N_G(d)$. Since edges $\{a, b\}, \{c, d\} \in E^*$ can share at most one vertex in common, it follows that, $d_G^I(v) \geq 3$, which is a contradiction as G is l -split, $l \leq 2$ and M is the labelled graph of G . Therefore, $|V^*| = |E^*|$. We also define the *Corresponding clique* set V^c of a vertex set $V' \subseteq I$ as follows. Corresponding to each vertex $u \in V'$, include exactly one vertex w in V^c such that $\{u, w\} \in E(G)$. Clearly, $V^c \subseteq C$ and $|V^c| \leq |V'|$. For a 1-split graph, $|V^c| = |V'|$. We now present some structural observations on $K_{1,4}$ -free split graphs.

Lemma 3. *Let G be a 3-split graph. G is $K_{1,4}$ free if and only if for every $u \in V_3$ and for every $v \neq u \in C$, $N_G^I(u) \cap N_G^I(v) \neq \emptyset$.*

Proof. Necessity: On the contrary, let us assume there exist $v \in C$ such that $N_G^I(u) \cap N_G^I(v) = \emptyset$. Since $d_G^I(u) = 3$, vertices $\{u, v\} \cup N_G^I(u)$ induces a $K_{1,4}$, which is a contradiction and the necessary condition follows.

Sufficiency: On the contrary, assume that G is not $K_{1,4}$ free and there exists a $K_{1,4}$ induced on $\{u, v, w, x, y\}$ with u as the centre vertex. No $K_{1,4}$ in G can have its centre vertex in the set I , since for any u in I , the set $N_G(u)$ is a subset of the set C and hence induces a clique in G . So every $K_{1,4}$ in G has its centre vertex in the set C particularly, $u \in C$. Since G is a 3-split graph, $d_G^I(u) = 3$. This implies that there exist at least one vertex of $K_{1,4}$, say $v \in C$, and $u \in V_3$. It follows that $N_G^I(u) \cap N_G^I(v) = \emptyset$, which is a contradiction and the sufficiency follows. This completes the proof of the lemma. \square

Corollary 1. *Let G be a $K_{1,4}$ -free 3-split graph. For any $v \in C$, the graph H induced on the vertex set $V(G) \setminus N_G^I(v)$ is a l -split graph for some $0 \leq l \leq 2$.*

On the contrary, suppose there exists a vertex $w \in C$ such that $d_H^I(w) = 3$. i.e., $w \in V_3$. It follows that $N_G^I(w) \cap N_G^I(v) = \emptyset$. By previous lemma, $N_G^I(w) \cup \{w, v\}$ induces a $K_{1,4}$, which is a contradiction. \square

Corollary 2. *Let G be a $K_{1,4}$ -free split graph and $v \in C$. If $N_G^I(v) = \{v_1, v_2, v_3\}$, then $N_G(v_1) \cup N_G(v_2) \cup N_G(v_3) = C$.*

Proof. By Lemma 3, for every $u \in C$, $N_G^I(v) \cap N_G^I(u) \neq \emptyset$. This implies that for every $u \in C$, $\{v_1, v_2, v_3\} \cap N_G^I(u) \neq \emptyset$. It follows that $N_G(v_1) \cup N_G(v_2) \cup N_G(v_3) = C$. \square

Now onwards, we investigate the Steiner tree problem on $K_{1,4}$ -free split graphs. For our discussions on Steiner tree problem, we fix the terminal set R to be I . Observe that l -split graphs for $l = 1, 2$ are $K_{1,4}$ -free split graphs. If G is a 1-split graph, then there does not exist a vertex $v \in C$ such that $d_G^I(v) \geq 2$. Therefore, the corresponding clique set of I forms the minimum Steiner set S of G where $|S| = |I|$. We shall now consider 2-split graphs for discussions. For a 2-split graph G , recall that the labelled graph M is such that $V(M) = I$, $E(M) = \{\{a, b\} \mid a, b \in I \text{ and there exist } v \in C \text{ such that } \{a, b\} = N_G^I(v)\}$. The following lemma gives the cardinality of a minimum Steiner set of any 2-split graphs.

Lemma 4. *Let G be a 2-split graph, and M be the labeled graph of G with $\alpha(M) = k$. Then any minimum Steiner set S of G is such that $|S| = |I| - k$.*

Proof. If M is a connected graph, then the minimum Steiner set in G corresponds to the minimum edge cover in M . For any graph M with maximum matching P , the cardinality of minimum edge cover is $|V(M)| - |P|$. Therefore, a minimum Steiner set S is such that $|S| = |V(M)| - |P| = |I| - k$. If M is not connected, let C_1, C_2, \dots, C_r be the components such that $C_1, C_2, \dots, C_i, i \leq r$ are non-trivial components with at least one edge and $C_{i+1}, C_{i+2}, \dots, C_r$ are trivial ones. For components C_1, C_2, \dots, C_i , we find the maximum matching P where $k = |P|$ and $Q \subseteq C$ be the corresponding vertex set of the matching P . Clearly, $|N_G^I(Q)| = 2|Q| = 2|P| = 2k$. Let Q' be the corresponding clique set of $I \setminus N_G^I(Q)$. From the definition of the corresponding clique set, $|Q'| \leq |I \setminus N_G^I(Q)|$. Note that, there does not exist two vertices $x, y \in I \setminus N_G^I(Q)$ such that $N_G(x) \cap N_G(y) \neq \emptyset$, otherwise it contradicts the maximality of P . Since there does not exist the possibility to have two such vertices $x, y \in I \setminus N_G^I(Q)$, it follows that $|Q'| = |I \setminus N_G^I(Q)|$ and the graph induced on $V(G) \setminus N_G^I(Q)$ is a 1-split graph. Therefore, $|Q'| = |I| - 2k$, and $I \setminus N_G^I(Q) \subseteq N_G^I(Q')$. It follows that the set $S = Q' \cup Q$ forms a Steiner set of G and $|S| = |I| - 2k + k = |I| - k$. \square

Lemma 5. *For any 2-split graph G , OPT Steiner tree problem is polynomial-time solvable.*

Proof. Finding the labeled graph M of G , incurs $O(n)$ effort where $n = |V(G)|$. Maximum matching P of M can be found in $O(n^{\frac{3}{2}})$ time. Note that the corresponding vertex set Q of P can be found in linear time. Similarly, the corresponding clique set also can be obtained in linear time. Therefore, the overall running time for finding the Steiner set is $O(n^{\frac{3}{2}})$ and OPT Steiner tree in any 2-split graph is polynomial-time solvable. \square

The following lemma characterizes a special 2-split graph constructed from a 3-split graph. Particularly, Lemma 6 gives an upper bound on the matching size of the labelled graph of a 2-split graph.

Lemma 6. *Let $G^1 = I^1 + C^1$ be a $K_{1,4}$ -free 3-split graph. For any $x \in V_3$, let G^2 be the graph induced on $V(G^1) \setminus N_{G^1}^I(x)$, and M be the labelled graph of G^2 . Then size of any maximum matching $\alpha(M) \leq 2$*

Proof. Recall from Corollary 1 that, G^2 is a l -split graph for some $0 \leq l \leq 2$. On the contrary, let $\alpha(M) \geq 3$. Let vertices $\{a, b, c, d, e, f\} \subseteq V(M)$ be those vertices participating in the matching of size at least 3 such that $\{u, v, w\} \subseteq C^1$ and $\{a, b\} \subseteq N_{G^1}^I(u)$, $\{c, d\} \subseteq N_{G^1}^I(v)$, $\{e, f\} \subseteq N_{G^1}^I(w)$ as shown in Figure 2. Clearly, from Lemma 3, $N_{G^1}^I(x) \cap N_{G^1}^I(u) \neq \emptyset$. Similarly, $N_{G^1}^I(x) \cap N_{G^1}^I(v) \neq \emptyset$ and $N_{G^1}^I(x) \cap N_{G^1}^I(w) \neq \emptyset$. We consider the following scenario.

Suppose $\{g, v\} \in E(G^1)$ and $\{g, u\} \notin E(G^1)$. Since $N_{G^1}^I(x) \cap N_{G^1}^I(u) \neq \emptyset$, without loss of generality, $\{h, u\} \in E(G^1)$. Observe that $\{u, v\} \cup N_{G^1}^I(v)$ induces a $K_{1,4}$. Therefore, $\{g, u\} \in E(G^1)$. Similar argument holds true for w and $\{g, w\} \in E(G^1)$. Since the clique C^1 is maximal, g is not adjacent to all vertices of C^1 ,

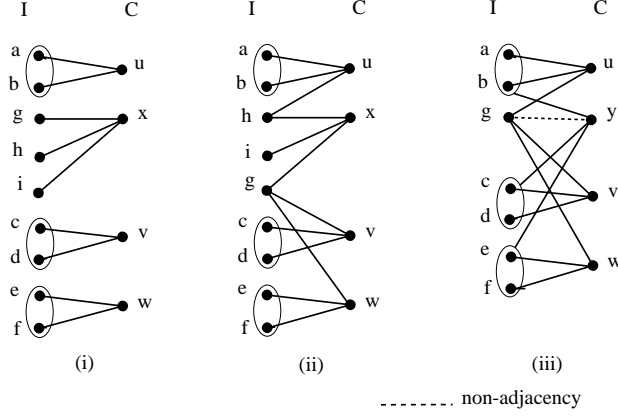


Fig. 2. An illustration for the proof of lemma 6

and therefore there exist $y \in C^1$ such that $\{g, y\} \notin E(G^1)$. Clearly, $N_{G^1}^I(y) \cap N_{G^1}^I(u) \neq \emptyset$, $N_{G^1}^I(y) \cap N_{G^1}^I(v) \neq \emptyset$ and $N_{G^1}^I(y) \cap N_{G^1}^I(w) \neq \emptyset$. Observe that in G^2 , $d_{G^2}^I(y) = 3$, and G^2 is not a l -split graph, $l \leq 2$. This is a contradiction to Corollary 1. It follows that our assumption $\alpha(M) \geq 3$ is wrong and therefore, $\alpha(M) \leq 2$. This completes the proof of the lemma. \square

We now present some structural observations pertaining to 3-split graphs.

Lemma 7. *For a $K_{1,4}$ -free 3-split graph G^1 , any Steiner set S of G^1 is such that $|S| \geq |I^1| - 5$.*

Proof. Observe that $V_3 \neq \emptyset$ as G^1 is 3-split. For any $v \in V_3$, G^2 is the graph induced on $V(G^1) \setminus N_{G^1}^I(v)$. By Corollary 1, G^2 is a l -split graph, $l \leq 2$. Let S^2 be the minimum Steiner set of G^2 such that $N_{G^2}^I(S^2) = I^2$ and $|I^2| = |I^1| - 3$. If M is the labeled graph of G^2 , then by Lemma 6, $\alpha(M) \leq 2$. Let $\{\{a, b\}, \{c, d\}\} \subseteq E(M)$ be the matching edges of a maximum matching in M . Observe that there exist two vertices $v_1, v_2 \in C^2$ such that $N_{G^2}^I(v_1) = \{a, b\}$, $N_{G^2}^I(v_2) = \{c, d\}$. Notice that for each vertex $w \in I^2 \setminus \{a, b, c, d\}$, there exist a vertex $u \in C^2 \setminus \{v_1, v_2\}$ in S^2 such that $\{w, u\} \in E(G^2)$. The graph induced on $V(G^2) \setminus \{a, b, c, d\}$ is a 1-split graph, $|S^2| \geq |I^2| - 4 + 2$ and it follows that $|S^2| \geq |I^1| - 3 - 2 = |I^1| - 5$. It can be concluded that $|S| \geq |I^1| - 5$ as $|S| \geq |S^2|$. This completes the proof of the lemma. \square

We below characterize $K_{1,4}$ -free 3-split graphs based on the cardinality of a minimum Steiner set. In particular, in Theorem 3, we characterize $K_{1,4}$ -free split graphs whose minimum Steiner set is $|I^1| - 4$, and in Theorem 4, we characterize $K_{1,4}$ -free split graphs whose minimum Steiner set is $|I^1| - 3$. To present Theorem 2 to Theorem 5, we fix the following notation. Let $G^1 = I^1 + C^1$ be a $K_{1,4}$ -free 3-split graph. For any $u \in V_3$, let $G^2 = I^2 + C^2$ be the graph induced on $V(G^1) \setminus N_{G^1}^I(u)$, and M be the labelled graph of G^2 . In Theorem 2, we present a stronger result of Lemma 7.

Theorem 2. *For a $K_{1,4}$ -free 3-split graph G^1 , any minimum Steiner set S of G^1 is such that $|S| \geq |I^1| - 4$.*

Proof. On the contrary assume that there exist a minimum Steiner set $S \subseteq C^1$ such that $|S| \leq |I^1| - 5$. Suppose that $S \cap V_3 = \emptyset$. Note that $V_3 \neq \emptyset$, say $u \in V_3$ and for every vertex $z \in S$, $N_{G^1}^I(z) \cap N_{G^1}^I(u) \neq \emptyset$ as per Lemma 3. i.e., for every $z \in S$, there exist an edge $\{z, i\} \in E(G^1)$, where $i \in N_{G^1}^I(u)$. The graph $G^2 = I^2 + C^2$ induced on $V(G^1) \setminus N_{G^1}^I(u)$ is a l -split graph, $l \leq 2$ by Corollary 1. Consider the Steiner set $S^2 \subseteq C^2$ of G^2 such that $N_{G^2}^I(S^2) = I^2$. Note that $|S^2| = |I^2|$ as G^2 is a l -split graph, $l \leq 2$ and for each vertex $w \in S^2$, $N_{G^1}^I(w) \cap N_{G^1}^I(u) \neq \emptyset$. Notice that $|I^2| = |I^1| - 3$ and $|S| \geq |S^2|$ implies that $|S| \geq |I^1| - 3$. This shows that $S \cap V_3 = \emptyset$ is not possible.

Next we shall consider the scenario $S \cap V_3 \neq \emptyset$. Consider the l -split graph, $l \leq 2$ G^2 induced on $V(G^1) \setminus N_{G^1}^I(u)$ where $u \in S \cap V_3$. Let the labeled graph of G^2 be M . For $S' = S \setminus \{u\}$ and $V' = I^1 \setminus N_{G^1}^I(u)$ note that $N_{G^2}^I(S') = V'$. Clearly, $|S'| = |S| - 1 \leq |I^1| - 5 - 1$ and $|V'| = |I^1| - |N_{G^1}^I(u)| = |I^1| - 3$. We now claim that

there exist at least 3 vertices say $\{v_1, v_2, v_3\} \subseteq S'$ such that $d_{G^2}^I(v_i) = 2, i = 1, 2, 3$ and $N_{G^2}^I(v_i) \cap N_{G^2}^I(v_j) = \emptyset, 1 \leq i \neq j \leq 3$. Suppose if there exist at most two vertices $v_1, v_2 \in S'$ such that $d_{G^2}^I(v_1) = d_{G^2}^I(v_2) = 2$ and $N_{G^2}^I(v_1) \cap N_{G^2}^I(v_2) = \emptyset$, then observe that $|V'| \leq |S'| + 2$. It follows that $|V'| \leq |I^1| - 6 + 2 = |I^1| - 4$, which is a contradiction as $|V'|$ is $|I^1| - 3$. Therefore, there exist at least 3 vertices $v_1, v_2, v_3 \in S'$ such that $d_{G^2}^I(v_i) = 2, i = 1, 2, 3$ and $N_{G^2}^I(v_i) \cap N_{G^2}^I(v_j) = \emptyset, 1 \leq i \neq j \leq 3$. Consider the labeled graph M of G^2 . There exist $\{a, b, c, d, e, f\} \subseteq I^2$ such that $\{a, b\} = N_{G^2}^I(v_1), \{c, d\} = N_{G^2}^I(v_2), \{e, f\} = N_{G^2}^I(v_3)$. It follows that $\{a, b\}, \{c, d\}, \{e, f\}$ forms a matching of size 3 in M which is a contradiction to Lemma 6. Therefore our assumption is wrong and $|S| \geq |I^1| - 4$. This completes the proof. \square

We show in Theorem 3 that the lower bound in Theorem 2 is tight.

Theorem 3. *For any minimum Steiner set S of G^1 , $|S| = |I^1| - 4$ if and only if $\alpha(M) = 2$.*

Proof. Necessity: If $S \cap V_3 = \emptyset$, then for every vertex $z \in S$, $N_{G^1}^I(z) \cap N_{G^1}^I(u) \neq \emptyset$. i.e., for every $z \in S$, there exist an edge $\{z, i\} \in E(G^1)$, where $i \in N_{G^1}^I(u)$. Similar to the proof of Theorem 2, it follows that $|S| \geq |I^1| - 3$. Therefore, $S \cap V_3 \neq \emptyset$. Let $u \in S \cap V_3$ and the graph G^2 induced on vertex set $V(G^1) \setminus N_{G^1}^I(u)$ is a l -split graph, $l \leq 2$ by Corollary 1. Let $S' = S \setminus \{u\}$. Clearly in $N_{G^2}^I(S') = I^2$. Note that $|S'| = |I^1| - 5$ and $|I^2| = |I^1| - 3$. This implies that there exist a matching of size at least 2 in M . From Lemma 6, $\alpha(M) \leq 2$. Therefore, $\alpha(M) = 2$.

Sufficiency: Let $\{a, b\}, \{c, d\} \in E(M)$ be the edges that form the matching of size 2 such that $\text{label}(\{a, b\}) = v_{ab}$ and $\text{label}(\{c, d\}) = w_{cd}$. Clearly, $d_{G^1}^I(u) = d_{G^1}^I(v) = d_{G^1}^I(w) = 3$ and $|N_{G^1}^I(X)| = 7$, where $X = \{u, v, w\}$. Let Y be the corresponding clique set of $I^1 \setminus N_{G^1}^I(X)$. Observe that $X \cup Y$ forms a Steiner set of G^1 , $|Y| = |I^1| - 7$ and $|X \cup Y| = |I^1| - 4$. This completes the proof. \square

Apart from the labelled graph M , we make use of one more labelled graph in Theorem 4, which is defined as follows. $H^2 = I_H^2 + C_H^2$ is the l -split graph, $l \leq 2$ induced on the vertex set $V(G^1) \setminus V_3$, and M^2 is the labeled graph of H^2 . Note that the two labeled graphs M and M^2 , are constructed differently. M is constructed on the vertex set $V(M) = I^1 \setminus N_{G^1}^I(u)$ whereas M^2 is the labeled graph on $V(M^2) = I^1$. We fix $S \subseteq C^1$ to be a minimum Steiner set of G^1 . The following theorem characterizes $K_{1,4}$ -free 3-split graphs with $|S| = |I^1| - 3$.

Theorem 4. *$|S| = |I^1| - 3$ if and only if one of the following is true.*

1. $S \cap V_3 \neq \emptyset$ and $\alpha(M) = 1$.
2. $S \cap V_3 = \emptyset$ and $\alpha(M^2) = 3$.

Proof. Necessity: If $|S| = |I^1| - 3$ then we come across the following two cases.

Case 1: $S \cap V_3 \neq \emptyset$.

Let $u \in S \cap V_3$ and $S' = S \setminus \{u\}$. Observe that, in G^2 , the l -split graph, $l \leq 2$ induced on the vertex set $V(G^1) \setminus N_{G^1}^I(u)$, $N_{G^2}^I(S') = I^2$. i.e., $|S'| = |S| - 1 = |I^1| - 3 - 1$ and $|I^2| = |I^1| - 3$. This implies that there exist a matching of size at least 1 in M . By Lemma 6, $\alpha(M) \leq 2$. Suppose $\alpha(M) = 2$, then by Theorem 3, $|S| = |I^1| - 4$. However, we know that $|S| = |I^1| - 3$ and therefore $\alpha(M) \neq 2$. We can therefore conclude that $\alpha(M) \leq 1$. If $\alpha(M) = 0$, then since G^1 is connected and $K_{1,4}$ -free, $|I^2| = 3$. In this case, $S = \{u\}$, $|S| = |I^1| - 2$. Therefore, it follows that $\alpha(M) = 1$.

Case 2: $S \cap V_3 = \emptyset$.

Since $S \cap V_3 = \emptyset$, S is a minimum Steiner set in G^1 and since H^2 is the induced on the vertex set $V(G^1) \setminus V_3$, S is also a minimum Steiner set in H^2 . Observe that if $\alpha(M^2) = k$, then the size of the minimum Steiner set in H^2 is $|V(I_H^2)| - k$ by Lemma 4. Since $V(I_H^2) = I^1$ we can conclude that $\alpha(M^2) = 3$.

Sufficiency: Case 1: $S \cap V_3 \neq \emptyset$ and $\alpha(M) = 1$

Let $u \in S \cap V_3$ and $\{a, b\} \in E(M)$ be the edge that forms the matching of size 1, such that $\text{label}(\{a, b\}) = v_{ab}$. Clearly, $|N_{G^1}^I(X)| = 5$, where $X = \{u, v\}$. Let Y be the corresponding clique set of $I^1 \setminus N_{G^1}^I(X)$. Observe that $S = X \cup Y$ forms a Steiner set of G^1 , $|Y| = |I^1| - 5$ and $|S| = |X \cup Y| = |I^1| - 3$.

Case 2: $S \cap V_3 = \emptyset$ and $\alpha(M^2) = 3$

Let $\{a, b\}, \{c, d\}, \{e, f\} \in E(M^2)$ be the edges that form the matching of size 3, such that $\text{label}(\{a, b\}) = v_{ab}$, $\text{label}(\{c, d\}) = w_{cd}$, $\text{label}(\{e, f\}) = x_{ef}$. Clearly, $|N_{G^1}^I(Y)| = 6$, where $Y = \{v, w, x\}$. Let Z be the corresponding clique set of $I^1 \setminus N_{G^1}^I(Y)$. Observe that $S = Y \cup Z$ forms a Steiner set of G^1 , $|Z| = |I^1| - 6$ and $|S| = |Y \cup Z| = |I^1| - 3$. This completes the proof. \square

Theorem 5. $|I^1| - 4 \leq |S| \leq |I^1| - 2$.

Proof. The lower bound is true by Theorem 2. Theorem 3, and Theorem 4 characterizes the 3-split graphs such that $|S| = |I^1| - 4$, and $|S| = |I^1| - 3$, respectively. We shall now look into the upper bound. Since G^1 is 3-split, there exist $u \in V_3$. Let Y be the corresponding clique set of $I^1 \setminus N_{G^1}^I(u)$. Observe that $N_{G^1}^I(Y \cup \{u\}) = I^1$ and $|S| \leq |Y| + 1$. Since $|Y| \leq |I^1| - 3$, it follows that $|S| \leq |I^1| - 2$. Therefore the theorem. \square

3.1 Polynomial-time algorithm to find a minimum Steiner tree

Using the structural results presented in Section 3, in this section, we shall present a polynomial-time algorithm to find a minimum Steiner tree in $K_{1,4}$ -free split graphs. Algorithm 3 finds a minimum Steiner set S of a given $K_{1,4}$ -free split graph G^0 with $R \subseteq V(G^0)$ being terminal vertices. Further, the minimum Steiner tree T is obtained using standard Breadth First Search on $G[R \cup S]$.

We shall now present a sketch of the algorithm and a detailed one is presented in Algorithm 3. As part of preprocessing, we prune the sets S_1, S_2 , and S_3 as defined in Section 2.1. Since G^1 is a $K_{1,4}$ -free split graph, G^1 is l -split, $l \leq 3$. We come across four cases as follows. If G^1 is a 0-split graph, then R is connected and the minimum Steiner set $S = \emptyset$. If G^1 is a 1-split graph, then the corresponding clique set of I^1 is a minimum Steiner set. If G^1 is a 2-split graph, then we find the labelled graph M of G^1 and the maximum matching P of M . Subsequently, we find the corresponding vertex set $Q \subseteq C$ of the matching P and the corresponding clique set Q' of $I \setminus N_G^I(Q)$. The minimum Steiner set is $S = Q \cup Q'$ (from Lemma 4). Given a 3-split graph, we perform a transformation to obtain a 2-split graph. We identify the size of a minimum Steiner set and the Steiner set with the help of the labelled graph associated with the transformed 2-split graph. Interestingly, based on the matching size, we get to identify the size of minimum Steiner set and the corresponding clique set helps us to identify the Steiner set. It is important to highlight the fact that if matching size is 1, we look at two different labelled graphs to identify the minimum Steiner set. The detailed algorithm is presented in Algorithm 3.

Algorithm 3 Compute_Steiner_Tree_ $K_{1,4}$ -free(G^0, R)

```

/* $G^0$  is the  $K_{1,4}$ -free split graph and  $R \subseteq V(G^0)$  is the set of terminal vertices */
1:  $G^1 = \text{Pruning}(G^0, R)$  i.e.,  $G^1 = G^0 \setminus (S_1 \cup S_2 \cup S_3)$ 
2: Initialize the output Steiner set  $S = \emptyset$ 
3: if  $G^1$  is a 1-split graph then
4:   Find corresponding clique set  $S$  of  $I^1$ 
5: else if  $G^1$  is a 2-split graph then
6:    $S = \text{Compute\_Steiner\_2-split\_graph}(G^1)$ 
7: else
8:    $S = \text{Compute\_Steiner\_3-split\_graph}(G^1)$ 
9: end if
10: Obtain the Breadth First Search tree  $T$  in the graph induced on vertices  $S \cup R$ .
11: Output  $T$ 

```

Algorithm 4 Compute_Steiner_2-split_graph(G)

```

/* $G$  is  $K_{1,4}$ -free and 2-split graph */
1: Initialize the Steiner set  $S^2 = \emptyset$ 
2: Construct the labeled graph  $M$  of  $G$ 
3: Find a maximum matching  $P$  in  $M$  and find the corresponding vertex set  $S^1$  of  $P$ 
4: Find the corresponding clique set  $S^2$  of the vertex set  $I \setminus N_G^I(S^1)$ 
5: Return Steiner set  $S^1 \cup S^2$ 

```

3.1.1 Proof of correctness of Algorithm 3 Step 1 of Algorithm 3 prunes the input graph G^0 to obtain G^1 and by Lemma 2, an optimal Steiner set of G^1 is also an optimal Steiner set of G^0 . If G^1 is a 1-split graph, then $|S| = |I^1|$ and our algorithm correctly computes such a Steiner set S in step 4. If G^1 is a 2-split or 3-split

Algorithm 5 Compute_Steiner_3-split_graph(G)

/ $G = I + C$ is $K_{1,4}$ -free 3-split graph*/*
1: Initialize Steiner set $S^2 = \emptyset$, $S^1 = \{u\}$ where $u \in V_3$ and edge set $P^1 = \emptyset$
2: **for** every vertex $v \in V_3$ **do**
3: Find the 2-split graph G^2 induced on $V(G) \setminus N_G^I(v)$ and the labeled graph M of G^2
4: Find a maximum matching P^* of M
5: **if** $|P^1| \leq |P^*|$ **then**
6: Update $P^1 = P^*$
7: Update $S^1 = \{v\} \cup$ corresponding vertex set of P^1 in G^2
8: **end if**
9: **end for**
10: **if** $|P^1| < 1$ **then**
11: Find the 2-split graph H^2 induced on $V(G) \setminus V_3$
12: Find the labeled graph M^2 of H^2 and a maximum matching P^2 of M^2
13: **if** $|P^2| = 3$ **then**
14: $S^1 =$ corresponding vertex set of P^2 in H^2
15: **end if**
16: **end if**
17: Find the corresponding clique set S^2 of the vertex set $I \setminus N_G^I(S^1)$
18: Return Steiner set $S^1 \cup S^2$

graph, then Algorithm 3 calls Algorithm 4, or Algorithm 5, respectively. Now we shall look into Algorithm 4 in detail. The algorithm finds the labeled graph M in Step 2. Note that Algorithm 4 in Step 3 finds a maximum matching P in M , and finds the corresponding vertex set S^1 of P such that $|S^1| = |P|$. Step 4 finds S^2 such that $|S^2| = |I^1| - 2|P|$. The Steiner set $S^1 \cup S^2$ is returned in Step 5 where $|S^1 \cup S^2| = |I^1| - |P|$, which is correct due to Lemma 4 and hence Algorithm 4 returns an optimum Steiner set.

In Algorithm 5, for every $v \in V_3$, we find G^2 and its labeled graph M in Step 3. A maximum matching on M is obtained in Step 4. Step 6 and 7 updates maximum matching P^1 and its corresponding vertex set S^1 found so far. Note that by Theorem 5, Steiner set S of G is bounded as $|I| - 4 \leq |S| \leq |I| - 2$. We can see the following cases.

Case (i) $|S| = |I| - 4$. By Theorem 3, $|P^1| = 2$ and it follows that $|S^1| = 3$ and $|N_G^I(S^1)| = 7$. Step 17 finds S^2 such that $|S^2| = |I| - 7$. Step 18 returns $S^1 \cup S^2$ where $|S^1 \cup S^2| = |I| - 4$.

Case (ii) $|S| = |I| - 3$. By Theorem 4, either $|P^1| = 1$ or $|P^2| = 3$. If $|P^1| = 1$, then $|S^1| = 2$ and $|N_G^I(S^1)| = 5$. Step 17 finds S^2 such that $|S^2| = |I| - 5$. Note that $|S^1 \cup S^2| = |I| - 3$. If $|P^2| = 3$, then $|S^1| = 3$ and $|N_G^I(S^1)| = 6$. Step 17 finds S^2 such that $|S^2| = |I| - 6$. Observe $|S^1 \cup S^2| = |I| - 3$.

Case (iii) $|S| = |I| - 2$. It follows that $|P^1| = 0$. Since we initialized S^1 with a vertex $u \in V_3$, $|S^1| = 1$ and $|N_G^I(S^1)| = 3$. Observe that $|S^2| = |I| - 3$ and $|S^1 \cup S^2| = |I| - 2$. This completes the case analysis and Algorithm 5 correctly computes a Steiner set of G . Therefore, Algorithm 3 correctly computes the minimum Steiner tree in Step 10.

3.1.2 Run-time analysis of Algorithm 3

Let n, m represents the size of vertex set, and the edge set, respectively of the input graph G^0 . We shall first analyze the run-time of Algorithm 4 and Algorithm 5 as Algorithm 3 invokes Algorithm 4 or Algorithm 5 at Steps 6, 8, respectively. For Algorithm 4, observe that creation of the labeled graph in Step 2 needs $O(n)$ effort as $|E(M)| + |V(M)| = O(n)$. Step 3 finds a maximum matching of M which can be done in $O(n^{\frac{3}{2}})$ time using general graph maximum matching algorithm [22]. Corresponding clique set in Step 4 can be found in $O(n)$ time. Therefore, the run-time of Algorithm 4 is $O(n^{\frac{3}{2}})$. Consider Algorithm 5, Steps 3 to 8 are iterated at most n times. Step 3 needs $O(n)$ effort. Finding a matching of M^2 in step 4 needs $O(n^{\frac{3}{2}})$ time. Steps 6, 7 incurs constant effort. Therefore, the iteration of Steps 3 to 8 involves $O(n^{\frac{3}{2}})$ effort. Note that Steps 11, 12 need $O(n)$, $O(n^{\frac{3}{2}})$, respectively and Step 14 incurs a $O(n)$ effort. Finding S^2 in step 17 can be done in $O(n)$ time. Overall, the run-time of Algorithm 5 is $O(n^{\frac{5}{2}})$.

Now we shall discuss run time of Algorithm 3. Pruning of verices in step 1 of Algorithm 3 takes $O(n \cdot \Delta)$ effort where Δ denotes maximum degree of the input graph G . Step 4 takes $O(n)$ time. Steps 6, 8 takes $O(n^{\frac{3}{2}})$ time, $O(n^{\frac{5}{2}})$ time, respectively. Step 10 incurs $O(n + m)$ time. Therefore the run time of Algorithm 3 is $O(n^{\frac{5}{2}})$. Thus, Steiner tree in $K_{1,4}$ -free split graph is polynomial-time solvable.

4 Steiner tree in $K_{1,5}$ -free Split Graphs is NP-complete

In the earlier section, we have presented a polynomial-time algorithm for Steiner tree in $K_{1,4}$ -free split graphs. In this section, we present the other half of the dichotomy, which is to show that Steiner tree in $K_{1,5}$ -free split graph is NP-complete. Interestingly, the reduction presented in [2] generates instances of $K_{1,5}$ -free split graphs. For the sake of completeness, we present our observations along with proofs. Towards this attempt, we recall the classical problem Exact 3 cover [23] which is a candidate NP-complete problem for our investigation.

Exact-3-cover(Z, T)

Instance: A Collection T of 3 element subsets of a set $Z = \{u_1, u_2, \dots, u_{3q}\}$.

Question: Is there a sub collection $T' \subseteq T = \{c_1, c_2, \dots, c_n\}$ such that for every $u_i \in Z$, $1 \leq i \leq 3q$ u_i belongs to exactly one member of T' ?

We recall the decision version of Steiner tree problem, restricted to $K_{1,5}$ -free split graphs.

Steiner tree(G, R, k)

Instance: $K_{1,5}$ -free Split Graph $G(V, E)$, Terminal Set $R \subseteq V(G)$, Integer $k \geq 0$

Question: Is there a set $S \subseteq V(G) \setminus R$ such that $|S| \leq k$ and $G[S \cup R]$ is connected?

Theorem 6. *Steiner tree problem in $K_{1,5}$ -free split graph is NP-complete.*

Proof. Steiner tree is in NP Given a certificate $S = (G, R, k)$, we show that there exist a deterministic polynomial-time algorithm for verifying the validity of the certificate S . Note that the standard Breadth First Search algorithm can be employed to check whether $S \cup R$ is connected. $|S| = k$ can be verified in linear time and therefore, overall certificate verification need $O(n + m)$ time, where $n = |V(G)|$, $m = |E(G)|$. Therefore, we can conclude that Steiner tree is in NP.

Steiner tree is NP-Hard An instance of Exact 3 cover(Z, T) is reduced to an instance of Steiner tree (G, R, k) problem as follows: $I = Z$, $C = \{v_i \mid c_i \in T\}$, $1 \leq i \leq n$ and $V(G) = I \cup C$. Informally, for every element $u \in Z$, create a vertex u such that $u \in I$. For every member $c_i \in T$, create a vertex v_i such that $v_i \in C$. $E(G) = \{\{v_i, v_j\} \mid v_i, v_j \in C\}$, $1 \leq i \neq j \leq n \cup \{\{v_l, u\} \mid v_l \in C, u \in I, \text{ and } u \in c_l\}$. $R = I$ and $k = \frac{|Z|}{3}$. In this reduction, $|V(G)| = |Z| + |T|$ and $|E(G)| = \binom{|T|}{2} + 3|T|$. The above construction is therefore polynomial to the size of input. We now show that instances created by this reduction are $K_{1,5}$ -free split graphs. On the contrary, assume that there exist a $K_{1,5}$ induced on vertices $\{u, v, w, x, y, z\}$. Note that at most two vertices (say u, v) from clique C can be included in the $K_{1,5}$. Clearly, $w, x, y, z \in I$ and without loss of generality, $d_G^I(v) = 4$. This implies that there exist a 4 element subset $c \in T$ corresponding to the clique vertex $v \in C$, which is a contradiction as all subsets are of size 3 in collection T . Therefore it follows that the reduced graph G is $K_{1,5}$ -free split graph. We now show that there exist an Exact-3-cover(Z, T) if and only if there exist a Steiner tree(G, R, k) in the reduced graph G on at most k Steiner vertices. For *Necessity*: If there exist $T' \subseteq T$, $|T'| = \frac{|Z|}{3}$ which covers all the elements of Z , then the set of vertices $S = \{v \in C \mid c \in T'\}$ where v is the corresponding vertex of c forms a Steiner set in G as $R = Z$. Also note that $|S| = \frac{|Z|}{3}$. For *Sufficiency*: If there exist a Steiner set $S \subseteq C$ in the reduced graph G on at most $k = \frac{|Z|}{3}$ Steiner vertices, then observe that for all vertex $v \in S$, $d_G^I(v) = 3$, $|S| = \frac{|Z|}{3}$ and $|N_G^I(S)| = |Z|$. It follows that there does not exist $u, v \in S$ such that $N_G^I(u) \cap N_G^I(v) \neq \emptyset$. Therefore, $T' = \{c \in T \mid v \in S\}$ where v is the corresponding vertex of c forms an exact 3 cover of Z . This completes the proof of the claim. We can conclude that Steiner tree problem is NP-complete in $K_{1,5}$ -free split graphs. \square

5 Conclusions and Future Work

We have presented an interesting dichotomy result that Steiner tree problem is polynomial-time solvable in $K_{1,4}$ -free split graphs and NP-complete in $K_{1,5}$ -free split graphs. This result is tight and it identifies the right gap between NP-completeness and polynomial-time solvability of Steiner tree in split graphs. Using the structural results presented here, an interesting direction for further research would be to explore the complexity of other classical problems which are NP-complete restricted to split graphs.

References

1. M. R. Garey, R. L. Graham, D. S. Johnson: The complexity of computing steiner minimal trees. *SIAM Journal on Applied Mathematics* **32**(4) 835–859 (1977)
2. Kevin White, Martin Farber, William Pulleyblank: Steiner trees, connected domination and strongly chordal graphs. *Networks* **15**(1) 109–124 (1985)
3. Stefan Vo: Steiner tree problems in telecommunications. In: *Handbook of Optimization in Telecommunications*, Springer US 459–492 2006
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: *Introduction to Algorithms* (3. ed.). MIT Press (2009)
5. Michael R. Garey, David S. Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York (1979)
6. M.C.Golumbic: *Algorithmic Graph Theory and Perfect Graphs*. Academic Press, New York (1980)
7. A.A.Bertossi, M.A.Bonuccelli: Hamiltonian circuits in interval graph generalizations. *Information Processing Letters* 195–200 (1986)
8. Michael R. Garey, David S. Johnson: The rectilinear Steiner tree problem is NP-complete. *SIAM Journal on Applied Mathematics* **32**(4) 826–834 (1977)
9. Haiko Muller, Andreas Brandstadt: The NP-completeness of steiner tree and dominating set for chordal bipartite graphs. *Theoretical Computer Science* **53**(2) 257 – 265 (1987)
10. J.M.Keil: Finding hamiltonian circuits in interval graphs. *Information Processing Letters* 201–206 (1985)
11. R.W.Hung, M.S.Chang: Linear-time certifying algorithms for the path cover and hamiltonian cycle problems on interval graphs. *Applied Mathematics Letters* 648–652 (2011)
12. B.S.Panda, S.K.Das: A linear time recognition algorithm for proper interval graphs. *Information Processing Letters* 153–161 (2003)
13. L.Ibarra: A simple algorithm to find hamiltonian cycles in proper interval graphs. *Information Processing Letters* 1105 – 1108 (2009)
14. Joseph A. Wald, Charles J. Colbourn: Steiner trees, partial 2-trees, and minimum IFI networks. *Networks* **13**(2) 159–167 (1983)
15. S.E. Dreyfus, R.A. Wagner: The steiner problem in graphs. *Networks* **1** 195–207 (1972)
16. Michael Dom, Daniel Lokshtanov, Saket Saurabh: Incompressibility through colors and IDs. In: *Automata, Languages and Programming*. Volume 5555. 378–389 2009
17. Naveen Garg: Saving an epsilon: A 2-approximation for the k-mst problem in graphs. In: *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*. 396–402 2005
18. Marcus Brazil, Ronald L. Graham, Doreen A. Thomas, Martin Zachariasen: On the history of the euclidean steiner tree problem. *Archive for History of Exact Sciences* **68**(3) 327–354 (2014)
19. M. Jones, D. Lokshtanov, M.S. Ramanujan, S. Saurabh, O. Suchy: Parameterized complexity of directed steiner tree on sparse graphs. In: *Algorithms - ESA 2013*. Volume 8125. 671–682 2013
20. Leonid Zosin, Samir Khuller: On directed steiner trees. In: *Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. 59–63 2002
21. D.B.West: *Introduction to graph theory* 2nd Edition. (2003)
22. S. Micali, V. V. Vazirani: An $O(\sqrt{VE})$ algorithm for finding maximum matching in general graphs. In: *IEEE Annual Symposium on Foundations of Computer Science*. 1980
23. R.M.Karp: Reducibility among combinatorial problems. In: *Proc. of a Symposium on the Complexity of Computer Computations*. 85–103 1972